心理学と数学的模型

——思考の非数量的模型を中心として——

高　田　洋　一　郎

§1　心理学と数学

心理学と数学との関係は，見方によっては Fechner にまでさかのぼることが出来るし，また，測定理論や検査理論の領域では，psychometrics の30年余に及ぶ伝統がある。しかし，知覚，学習といったような，実験心理学の中心的な問題に対して数学的模型化が試みられるようになったのは第二次大戦中のアメリカにおいてであって，戦後，次第にその数が増大した。特に最近10年ほどの間に，アメリカの心理学者によって試みられた数学的模型は可成りの数にのぼり，しかもその数は，年を追って増す傾向がみられる。

一方，このような数学的模型の試みに対して，基本的な批判が考えられないこともない。明確に表明された反対意見というのは現在のところ存在しないようであるが，アメリカの心理学においても，数学的模型が全面的に受け入れられているとは思えないし，また，ヨーロッパの心理学では殆んど皆無であることも注目をひく（数少ない例の一つとして，Piaget（1953）の論理記号を用いた思考発達の模型があげられるが，彼の論理数学は可成り独得なもので問題点も甚だ多いように思われる）。わが国においても，心理学で数学を用いることに対する不信——ある場合には極めて正当な不信——が底流としては強く存在するように思われる。

このような批判——というよりむしろ疑惑——は，ある意味で十分根拠のあることである。自然科学での数学的模型乃至理論は，いわば，科学の内部からの必然的要求によって生じている。つまり，日常の言葉による概念では，もはや現象の分析や一般化がどうしても出来なくなったときに，日常言語より一段高い正確さと抽象性を具えた数学的言語が要請されたといってよい。そして，自然科学，特に物理学は，その研究の歴史において，単に既成の数学を適用するにとどまらず，屢々自ら新しい数学を創り出して来た。

一方，心理学では，数学利用の歴史が浅いことが大きな原因ではあろうが，未だかって新しい数学が創り出されたことはないし，また，既成の数学の利用

の仕方も，その領域における探究の結果，必然的に要請されたというより，むしろ試みに，時には可成り便宜的に，数学的模型化を行ってみたという程度のものが多い。

数学的理論は，常に現象の抽象化であり，抽象化とは，現象を規定していると考えられる要因のうち，あるものを採って他を無視することであろう。数学的理論は，日常の言葉による記述とちがって，現象のいろいろなニュアンスをひろく汲みつくすには余り適していない。それだけに，数学的抽象化にあたっては，現象を規定している最も本質的な要因を鋭くえぐり出す努力が強く要請されるであろう。そうでないと，数学的理論は，正確ではあっても内容は空虚なものになってしまう。心理学でも数学的模型は，日常の言語による記述が必然的に伴う意味の曖昧さ，理論の不厳密さを排し，厳密な予測を可能にしたという意味では，疑いもなく大きな貢献をなしてきた。しかし一方，数学的模型につきものの抽象化によって，心的現象の最も本質的な要因がこぼれ落ちてしまったのではないかという疑いが生ずるのは自然であり，また，この疑いを一歩進めれば，心的現象の本質は数式によっては汲みとれないものであるという主張になるであろう。実際，心的現象のような複雑な現象に既成の数学——それは主として自然現象の解明に関連して発達してきたものである——を無理に適用しようとすれば，多少とも強引な抽象化，時には，単に数学の技術的な理由による抽象化を行わねばならないことが多い。例えば，統計的学習理論で用いられている数学的模型は線型差分（微分）方程式であるが，非線型式ではなくて線型式を用いた理由について，模型の作成者は次のように述べている（Bush & Mosteller (1955), p. 46)。「われわれが線型演算子を選んだ理由は主として数学的なものである——すなわち演算子を線型とすることによって理論を処理し易い形にするためである………」このようなアプローチの仕方は，見方によっては，余りにも便宜的であると批判されよう。もっとも，あらゆる理論は近似であると考えれば，時には技術的な理由による抽象化もまた止むを得ないのであって，自然科学でも，非線型現象の解析は漸く最近始まったばかりであることを思えば，上のような批判は時には正当な批判であっても，心理学で数学を用いることへの基本的な不信とはなり得ないと言えるかもしれない。基本的な不信は，やはり，数式によっては最も本質的な要因を捉えることができないという疑惑に発するものであろうか。

このような疑惑は，確かに心理学における数学的模型の現状だけに視野を限定すれば，ある意味で尤もな疑惑である。しかし一方，数学的模型の将来の可

能性をも展望するとき，かかる疑惑を根拠づける十分な証拠があるとも思えない。ただ，一つ考えられるのは，心的現象の数量化ということである。数学は，ある面では数量の科学であり，実際，現在までの心理学の数学的模型は殆んどが数量的模型である。いうまでもなく，心的現象の数量化には，数々の困難が伴う。人間科学の中で，心理学が一番，数量化に関し困難な問題をかかえていると言えるかもしれない。心理学は，長い間の苦心を通じて，数々の数量化の技術を発展させてきてはいるが，数量化の基本に関しては，未だ必ずしも意見の一致をみていないと言える。そして，上に述べたことに関連するが，比較的容易に数量化できる側面だけを捉えて模型化するときには，本質的に重要な要因が落ちてしまうのではないかという疑惑が確かに濃厚に生じるのである。

このような事情を考えるとき，最近試みはじめられた非数量的模型——計算機のプログラムとして書かれた模型 (computer simulation models)——は，注目に値すると思う。このような試みはまだ始まったばかりであり，最近出版された数理心理学のハンド・ブック (Luce *et al* (eds): Handbook of mathematical psychology. vol. 1, 1963. 2巻まで刊行されており，全3巻からなる予定) の中でも，非数量的模型については， Newell & Simon による一章があるにすぎない。しかも，著者ならびにこの本の編集者の意見によれば，この章がハンド・ブックに加えられたことも，数理心理学の伝統からいえば歴史的偶然（最近数年間，心理学での計算機利用の急速な増大にもとづく）であると言う。しかし乍ら，心理学の現状において数量的模型のもつ数々の困難を考えるとき，非数量的模型は大きな可能性を示唆していると言えるし，また，方法論的にも極めて興味ある問題を提起していると思う。問題は，これが数学的模型と呼べるかということである。

ここで，数学的とはどういうことかということについて論ずることは，専門家でない私のよくするところではない。しかし現代の数学が公理主義をたてまえとしていることは，よく知られたことであろう。公理というものの捉え方は，周知のように，論理主義，直観主義，形式主義という，数学基礎論の立場の何れをとるかによって相異するが，現代数学の大勢を占める形式主義の立場にたてば，公理系とは次のようなものであるといわれる（赤 (1963)）。公理系は，それ自体としては無定義な，あるいくつかの対象と，いくつかの函数と述語とをそれぞれあらわす記号から構成される。ここで函数とは，一つあるいは二つ以上の対象に対象を対応させる操作を意味し，述語とは，一つの対象の性質を示したり，二つ以上の対象の関係を示したりする記号を意味する。対象も函数

も述語も，公理系の中では内容的には定義されず，ただこれらを記号として操作する一連の規則によってのみ規定されている。これらの記号，ないし記号間の関係をあらわす記号，および若干の論理記号の集まりが，公理系をつくる。勿論，数学を科学に適用する際には，公理系の中の記号の解釈が行われなければならない。しかし，公理系はあらかじめ一つの解釈を強制するものではなく，どのような解釈でも，それが公理系の規則に反していない限り，全く自由であるところに数学というものの特色がある。

以上のような捉え方では，公理系の記号の中には数をあらわす記号がなければならぬというようなことは何等前提とされていないことが明らかであろう。実際，現代の数学では，数や，数の函数や，数に関する述語をあらわす記号以外の記号を扱う領域が益々拡がりつつある。

そこで，ある模型が数学的模型であると言い得るためには，次の条件がみたされねばならないことになる。 1) 明確な記号の体系であること。 2) 記号を操作する規則が明確に規定されていること。ところで，後に述べる非数量的模型は正にこの条件をみたしているといえる。何故なら，これらは計算機のプログラムとして書かれたものであるが， 1) と 2) の条件をみたすことなくしては計算機のプログラムとすることが出来ないからである。もっとも 1) と 2) とは，数学的模型のいわば，必要条件であって，充分条件ではない。充分条件としては，更に模型とは何かということも考慮しなければならないであろう。この点は最後の節で触れることとして，ここではただ，計算機のプログラムの原理は数学基礎論のある分野と密接に関連していることを指摘しておくにとどめる。

以下，非数量的模型の一例として，思考の模型について考察してみたい。最初に，思考を模型化するということは一体どういうことであるのかという問題をとりあげる。

§2　論証について

思考というものの一つの重要なはたらきとして論証ということをあげることが出来よう。論証とは，いうまでもなく，ある前提からある終結を導くことであるが，どのような導き方を論証というのかということを，厳密な論証を生命とする数学についてまず見てみよう。前述の形式主義の立場に立てば，一般に論証というものは大略次のように形式化される（詳しくは前掲赤（1963）参照）。前述のように，公理系とは，その形式だけに着目すれば，記号の集まりに他な

らないから，適当に論理的な術語（“あるいは”，“そして”，“ならば”等の言葉）をも記号化することによって，公理系を構成する各公理を記号の列としてあらわすことが出来るはづである。かかる記号の列を命題と呼ぶことにし，いま，ある公理系が n コの命題，A_1，$A_2, \cdots\cdots, A_n$ からなるとする。この公理系の中で証明できる定理はまた，この公理系の記号を用いたある列であらわすことが出来るはづである。さて，ある定理 A を公理系（A_1，$A_2, \cdots\cdots, A_n$）の中で証明（論証）するとはどういうことかというと，それは，この公理系の中で最も自明な前提から出発して，普遍的に許されると思われる推論の規則のみを用いて，命題 A に到達することであると考えられる。いま，$A_1, A_2, \cdots\cdots, A_n \rightarrow B_1, B_2, \cdots\cdots, B_m$ という記号で，命題 A_1，$A_2, \cdots\cdots, A_n$ を前提として命題 B_1，$B_2, \cdots\cdots, B_m$ のうちの少くとも一つが成立するという主張をあらわすとすれば，与えられた公理系の中で最も自明な前提として例えば，A_1，$A_2, \cdots\cdots, A_n \rightarrow A_1, A_2, \cdots\cdots, A_n$ をあげることが出来よう。そのように考えると，すべての推論の規則は $\Gamma \rightarrow \Theta$（$\Gamma$ 及び Θ は，それぞれ命題の列）という形をした式を同様な形をした式に変形する（書きかえる）規則としてあらわすことが出来る。このような式の変形規則を数え上げてみると，20前後あれば十分であることが示されている。そこで，命題 A の証明とは，式 $A_1, A_2, \cdots\cdots, A_n \rightarrow A_1, A_2, \cdots\cdots, A_n$ から出発し，普遍的に許される変形の規則のみを用いて上式を変形することを有限回繰返して，式 $A_1, A_2, \cdots\cdots, A_n \rightarrow A$ を導出することであるといえよう。

以上は，公理化された数学において，論証というものがもつ，あるいはもつべき，形式の概略であるが，数学以外の領域における論証——諸科学における論証や日常の思考における論証——についてはどうであろうか。そもそも，数学における論証というものの上述のような形式化は 周知のように，数学の基礎領域に数々の矛盾が発見されるに及んで，いかにすれば矛盾をはらむ危険性のない論証が可能かという深刻な反省から生まれたものであった。実際，上述のように論証を形式化すれば，ある公理系（$A_1, \cdots\cdots, A_n$）が矛盾をふくむとは，$A_1, \cdots\cdots, A_n \rightarrow A_1, \cdots\cdots, A_n$ を変形して，$A_1, \cdots\cdots, A_n \rightarrow A \wedge \bar{A}$（$A \wedge \bar{A}$ は，A であり，そして A でないという命題をあらわす）という形の式が導出可能なことである，と極めて明確に表示することが出来る。したがって，このような形式化は，単に数学という特殊領域にかぎらず，およそ論証というものが厳密（矛盾を含まない）であるために持つべき論理構造をえぐり出したものであると言えよう。実際，記号論理学では，あらゆる論証を上述とほぼ同様な形に形

式化して捉えているのである（例えば Kleene (1952) 参照）。細かい議論に立ち入る余裕はないが，論証というものは一般に，前提といわれるある記号配列から出発し，これを普遍的に認められると思われる変形の規則を用いて次々と書きかえてゆくことによって，有限のステップで，終結と呼ばれるある記号配列に到達することである，と言って差支えないであろう。

さて，以上は，論証というものが一般にもつべきであると考えられる形式について述べたものであるが，思考の心理学にとって興味があるのは，具体的な論証が行われる姿にある。すなわち，ある具体的な前提式を次々と書きかえて，ある具体的な終結式に到達するためには，許された変形の規則をある順序で用いなければならないが，論証の各ステップでどの規則を用いるべきかの選択，あるいは，ある式を変形する際に，式に含まれる記号の中のどれを書きかえるかの選択，このような選択の系列が一つの具体的な論証を形づくる。この選択を規定する機構が一体どのようなものであるかということが，論証の心理学の重要な関心事であり，また，かかる機構を模型化したものが論証的思考の模型となるであろう。しかし，このことについて述べる前に，上述の議論と関係し，同時に，計算機のプログラミングの原理とも深く関連している Turing machine というものについて触れなければならない。

§ 3　Turing 機械と問題解決

イギリスの数学者 Turing は，1936—7年に，ある automatic machine の構想を発表した（Turing (1936—7)）。これが後に，Turing 機械（Turing machine）と呼ばれるようになったものである。もっとも，Turing 自身は自動機械の設計に興味があったわけではなく，数学基礎論のある種の問題を解くに際して，論証の過程を出来る限り簡単で明確な要素の組合せとして表示する必要を感じ，その目的のために，ある機械化したメカニズムを仮想したのである。しかしその後，後述のように，プログラム内臓型計算機が開発されるに及んで，Turing の仮想したメカニズムはまさに計算機の原理を示したものであることが明らかになったのである。Turing 機械というのは，丁度録音中の録音器のヘッドがテープに吹きこまれている音を消しながら，新しい音を書きこんでいくように，テープに書きこまれた記号を次々と書きかえてゆく装置であるが，正確に言うと次のような動作特性で定義される。

1.　Turing 機械は無限に長いテープをもっている。このテープはマス目に分かれていて，一つのマス目に一つの割で記号を書きこむことが出来る。テー

プ上の記号の配列が Turing 機械の入力である。

2. Turing 機械は録音器のヘッドのように，テープの各マス目を一つづつ走査しながら，そこに書き込まれた記号の書きかえを行っていくのであるが，その出力は二通りに分けて考えることが出来る。その一つは記号の書きかえであって，各ステップで Turing 機械はそのとき走査しているマス目の記号を消して，同じマス目に新しい記号を書き込む。その二は，テープ或いはヘッドの位置づらしであって，各ステップで記号の書き込みが終ったら，テープを一マス分だけ左へ動かす（ということは，ヘッドが一マス分右へ動く，すなわち，次に走査するマス目が右隣りに移ることに等しい。この移動を R であらわそう）か，或いは一マス分右へ動かす（L）か，或いは全然動かさない（次に走査されるマス目が前と同じになる。N）か，以上3通りの位置づらしの何れかが行われる。

3. テープにあらかじめ書き込まれる記号，および Turing 機械の書きかえによって生ずる記号はすべて，あらかじめきめられた1組の有限個の記号（それは任意にきめてよい）の中の何れかである。すなわち，この1組の記号を $S_1, \cdots\cdots, S_n$ とすると，Turing 機械の行う各ステップでの書きかえは，$S_i \to S_j$（S_i を S_j に書きかえる。$i, j = 1, \cdots\cdots, n$）であらわされる。ここで，$i = j$ であってもよいから，書きかえの中には「書きかえをしない」ことも含まれるわけである。更に，$S_1, \cdots\cdots, S_n$ の中には空白をあらわす記号（空記号）も許すことにする。すると，あるマス目の記号を空記号に書きかえるとは，そのマス目の記号を消し去ることを意味することになる。

4. Turing 機械が各ステップで書き込む記号 S_j は，そのとき走査しているマス目の記号 S_i が何であるかということと，そのときの機械の状態 q_k によって一義的にきまる。Turing 機械のとり得る状態の数は有限である（$q_1, \cdots, q_m$）。同様に，各ステップでの位置づらし（R か L か N か）も，そのとき走査した記号 S_i の種類と，そのときの状態 q_k によって一義的にきまる。一方，各ステップでの機械の状態は，その一つ前のステップの入力記号と状態によって一義的にきまる。

以上の定義によれば，Turing 機械の動作の仕方は，入力記号 $S_1, \cdots\cdots, S_n$ を行にとり，状態 $q_1, \cdots\cdots, q_m$ を列にとって，S_i 行 q_k 列欄に，次の三つの記号を書き込んだ表によって完全にあらわされることになる。三つの記号とは，入力記号が S_i で状態が q_k の時の出力記号と位置づらしをあらわす記号（R, L, N）および，次のステップでの状態をあらわす記号である。実際，いまある記

号配列がテープ上に与えられ，更に，機械の最初の状態と最初の位置がきめられたとすると，最初の入力記号がきまるから，上述の表（これを Turing 機械の動作表という）を引いて，最初の書きかえがきまる。また，同じ表で位置づらしと次の状態がきまるから次の入力記号と次の書きかえもきまる――以下同様にして次々に書きかえと機械の状態とがきまっていく。このように，一つの動作表を与えれば，一つの Turing 機械が完全に決定されるから，Turing 機械とはかかる動作表のことであるといってもよい。いま，動作表のある欄で出力記号が入力記号と一致し，次の状態が前の状態と同じであり，しかも位置づらしが N であるとしたら，これは何を意味するであろうか。明らかに Turing 機械は一たんその入力，その状態に立ち至ると，それ以後は，テープにも機械にも何等変化が起きないことを意味する。かかる状態を，Turing 機械の停止と呼んでよいであろう。テープに任意の記号配列を書き込んで Turing 機械の作動を開始したとき（ただしテープの記号はすべて Turing 機械の動作表にあげられた入力記号の組に含まれるとする），Turing 機械は，次々と最初の記号配列を書きかえていって，有限のステップで停止するか，あるいは永久に停止しないかの何れかである。

さて，前節に述べたように，一般に論証とは，前提という記号配列を一組の推論の規則を用いて次々と書きかえていって，有限のステップで終結という記号配列に到達することであり，そして論証の具体的手続きとは，推論の規則という記号書きかえの規約を，ある順序で選択し，適用することであった。したがって，一つの具体的な論証手続きは，常に一つの Turing 機械の動作表として表わすことが出来るはずである。すなわちこの機械に，前提を書き込んだテープを入れてやれば，機械は有限のステップで停止し，そして停止したとき，テープ上に書かれている記号配列は終結を表わすことになる。

このように一つの論証手続きは一つの動作表に表わすことが出来るが，逆に一つの動作表（一つの Turing 機械）のなし得ることは一つの具体的な論証よりも広い。何故なら，一つの動作表は一般に多種多様な記号配列に適用可能（有限回のステップで停止すること）であり，したがって，多種多様の前提式（その記号配列）を出発点の入力とすることが出来るからである。このことは別の言葉で言えば，Turing 機械には一種の記憶作用があるので，条件に応じて異なった動作をすることが出来るということに他ならない。前述のように，Turing 機械は有限個の異なった状態をとることが出来，そして出力（書き込む記号と位置づらし）は，入力が同じでも，機械の状態によって変化し得る。

そして各ステップでの状態は一つ前のステップでの入力と状態の函数であり，一つ前の状態は，もう一つ前の入力と状態の函数である。したがって，第nステップにおける機械の状態は最初の機械の状態と第$n-1$ステップまでの入力の系列の函数であることになるが，一方，第$n-1$ステップまでの入力の系列は，最初の記号の配列，最初の位置と状態，第$n-1$ステップまでの出力の系列に依存している。ということは，Turing 機械は各ステップにおいて，それまでの動作の結果およびテープと自己の初期状態と初期位置とを完全に記憶していて，かかる記憶を背景としてその動作を決定出来るということを意味する。したがって，論証について言えば，一つの Turing 機械は，前提およびその操作（式の変形）の結果の変化に応じて，異なった一連の動作を営むことが出来るのである。このように，条件づきで変化し得る論証手続きが適用可能な問題は，一般に一つの型の問題群を構成するであろう。ある型の問題の一般的解法の手続きを algorithm というが，この言葉を用いれば，一つの Turing 機械の動作表は一つの algorithm を表わし得ると言ってもよい。

実は，一つの動作表の表示し得るものは一つの algorithm よりも更に広いのである。論証というのは一種の問題解決行動であるとみることが出来るが，人間の行う問題解決行動は，もちろん論証に限らない。しかし，一般に問題解決行動というものが，出発点の状態（初期状態）と目標の状態とが与えられたとき，ある許された範囲の操作によって目標に到達することであるとみなされるならば，問題解決の過程は，初期状態から目標状態への状態変化の系列としてあらわされるであろう。そこでもし，適当な記号を用いて，初期状態と目標状態とをそれぞれ記号の列として表わし，かつ，許された操作を記号列の変形の規則に対応させることが出来るならば（かかる記号化は状態およびその変化が明確に表示出来る場合には，常に可能である），一般に問題解決行動というものを記号列の変形（書きかえ）として捉えることが出来ると言える。この場合「ある型の問題に対して一般性をもった解決行動」――初期状態の変化と操作の結果の変化に応じて，ある範囲で変化し得る問題解決行動――は，「条件に応じて変化し得る記号書きかえの手順」に対応することになるであろう。ということは，一つの Turing 機械の動作表は，一つの型の問題に対する行動の手順を表示し得るということである。

一般的な問題解決行動においては，与えられた初期状態と目標状態に対し，必ず解決（目標達成）を保証するような行動の手続きが存在するとは限らない。このような場合人間は，解決の保証はないが，しかし，しばしば有効であ

ることが知れている行動手続きを用いるであろう。このように，ある型の問題に対し，解決の保証はしないが有効であることが知れている一般的な行動手順を algorithm に対して heuristic というが，この言葉を用いれば，Turing 機械の動作表は algorithms のみならず heuristics をも表示し得ると言える。

問題解決の heuristics については次節で触れることとして，ここではあと，Turing 機械と計算機のプログラムとの関係について簡単に触れてみたい。前述のように，一つの algorithm は一つの型の問題に対する一般的解法の手順を与える。科学の探究の歴史は，一面において，絶えず新しい型の問題を提起し，それに対する解法を探るものであったと同時に，他面，特殊な型の問題の間に共通の原理を求め，特殊な型の問題をより一般的な型の問題に包含することによって，より一層一般的な問題の解法を求めるものであったと言える。そこで自然に起って来る疑問は，多種多様な型の問題，出来ればあらゆる型の問題を解く algorithm はつくれないであろうかということであろう。Turing の考えた万能機械（universal Turing machine）の構想は，この疑問に対する一つの答を与えている。万能 Turing 機械とはあらゆる algorithm を行うことの出来る動作表をもった一つの Turiug 機械である。あらゆる algorithm を行うというと，一見大変なことのようであるが，原理的には頗る簡単なことで，万能 Turing 機械は，任意の Turing 機械のものまねをすることが出来るのである。ある Turing 機械に別の Turing 機械のまねをさせるには，前者のテープに後者の動作表をあらかじめ書きこんでおいて，与えられた記号配列を走査する際に，各ステップでこの動作表を参照して動作を決めるようにしてやればよい。そして，一寸した工夫によって，このようなものまねの動作表をつくることが出来るのである。ものまね動作表をもった Turing 機械が，万能Turing 機械に他ならない。ただし，ここで注意を要するのは，万能 Turing 機械にはあらゆる algorithm が実行出来るという事は，決して，あらゆる問題が万能 Turing 機械で解けることを意味しないということである。万能 Turing 機械は algorithm が存在する問題に対しては必ず解決を与える。しかし，algorithm が存在しないことが証明されている問題がいくつかあるのであって，例えば，上述の，あらゆる型の問題を一般的解法で解くという問題がそれである。

しかし，ともかく，万能 Turing 機械は，algorithm が存在する限り，それを実行出来るのであるから，甚だ強力であるに違いないが，現代の電子計算機，正確に言えば，プログラム内臓型逐次制御自動計算機は，万能 Turing 機械の構想を実現したものに他ならない。すなわち，万能 Turing 機械のテープは計

算機の記憶装置に相当し，テープ上に書き込まれた出発点の記号配列は計算機への入力データーに対応し，テープ上の任意の algorithm は計算機のプログラムにあたる。万能 Turing 機械が万能（上述の意味で）であるのは，そのテープ上に書き込まれた任意の algorithm を実行することが出来るからであるが，同様に，計算機が汎用（all purpose）であると言われるのは，記憶装置の中にデーターと共に任意のプログラムを貯えて（プログラム内臓）これを実行出来るからに他ならない。現代の計算機は汎用情報処理機であると屢々言われるが，その意味は単に数の計算に限らず，任意の algorithm，任意の heuristic が実行出来るということである（ただし，Turing 機械では無限に長いテープを仮定しているが，計算機の記憶容量は常に有限である。したがって，与えられた algorithm 乃至 heuristic に相当するプログラムと入力データーとをあわせたものが，与えられた計算機の記憶容量を超えない，という但し書きが常に必要である）。次節で述べる思考の非数量的模型は，かかる計算機の機能を利用したものに他ならない。

§4　heuristics について

論証の algorithm を求めるという事は，単に数学にとどまらず，諸科学においても極めて重要である。科学の理論というものは，いうまでもなく，個別的な規則性の単なる寄せ集めではなく，ある種の法則性が，より基本的な法則から論証されるという形で体系化されたものでなければならない。例えば，ケプラーの法則は万有引力の法則と運動の三法則から論証され，ボイル・シャルルの法則は気体分子の熱運動の法則から論証されるが如きである。科学者は現象の規則性を発見したとき，その規則性をその領域で基本的とみなされている法則から論証しょう（algorithm をつくろう）と努めるであろう。このような事情は，数学でも諸科学でも本質的には違いがない。

思考の心理学にとって興味があるのは，algorithm そのものよりも，むしろ algorithm がどのようにしてつくられるかということ——algorithm 発見の思考過程である。何故なら，与えられた algorithm を与えられたデータに適用することは，万能 Turing 機械の行うものまねに類する事であって，広義の思考活動の一部をなすものであっても，思考の本質とは到底見なし得ないからである。もっとも，既知の algorithm の中から，与えられたデータに最も適したものを選択して用いる行動は，いわゆる創造的思考ではなくとも，可成り重要な思考活動ではあろうが，この点については後に触れる。

前述のように，あらゆる algorithms の発見を可能ならしめる algorithm というものは存在しない。したがって，algorithm 発見の一般的方法は，必然的に heuristics にならざるを得ない。もっともこのように言うと次のような疑問が生じるかも知れない。前に述べたように，一般に問題解決というものが，与えられた初期状態を逐次変形して目標状態を実現する事であるならば，許された変形規則をあらゆる順序で悉く試みてみればよい――到達可能な目標なら，変形状態の何れかにあらわれるはずである。つまり，しらみつぶし，あるいは試行錯誤という algorithm があるのではないか，という疑問である。しかし，このような方法は一般的な algorithm にはなり得ない。何故なら，一般に，与えられた変形規則を悉く，あらゆる順序で試みることは有限のステップでは不可能であり，したがって，しらみつぶしの方法は一般に問題解決の論理的保証を与えない（Turing 機械が停止しない）からである。ただ，問題によっては，あらゆる変形規則のあらゆる順列が有限である場合もあって，このとき，しらみつぶしの方法は，当然，一つの algorithm となる。例えば，チェスのようなゲームに勝つ一般的方法という問題がそれであって，一局のゲームは必ず有限のステップで勝敗が決するから（双方手づまりによる引分けがあり得る時には，上の「勝つ一般的方法」というのを，「負けない一般的方法」と言いかえればよい），局面の変化（チェス盤上の白駒と黒駒の配置の変化。その数は有限である）の順列も有限である。したがって，プレイの開始から終了までのあらゆる局面の変化をしらみつぶしに調べれば，先手か後手か何れかの必勝の algorithm がつくられるはづである。しかしながら，チェス必勝の戦略はまだつくられたことがない。何故かというと，理論的には確かに上述の通りであるが，実際には，あらゆる局面の変化の数というのが，チェスのような比較的簡単なゲームでも，想像を絶する大きさになるのであり，したがって，それをしらみつぶしに調べることは実際問題として不可能であるからである。Shannon の概算によると，最高速度の計算機をもってしても，必勝のalgorithm を発見するには 10^{100} 年近くかかるという（Shannon & McCarthy (1956)）。これは途方もなく長い時間であって，計算機の速度がいかに急速に向上しても，しらみつぶしの方法で algorithm をつくるということは全く問題にならないと言える。更に，これ程の大きさでない場合でも，algorithm の発見に必要な動作の数が，ある程度以上になると，計算機でも誤動作の可能性が無視出来なくなるということを考えねばならぬ。一般に algorithm というものは，それが正しい限り，問題解決の完全な保証を与えてくれるが，一ケ所でも誤り

があると何の役にも立たないのである。いわば「当らずといえども遠からず」式の heuristic と対蹠的であると言える。

現在までに，チェスを指すプログラムがいくつかつくられているが，それらは何れも何らかの heuristics を用いている。例えば，局面のあらゆる変化ではなくて，数手先の変化まで読んで，そのときの局面の有利さを駒の数や位置などから評価し，最も有利な変化をもたらす手を選択する，というようなプログラムである（これらのプログラムについては Feigenbaum & Feldman (1963) に詳しい紹介がある）。いうまでもなく，これは人間がチェスを指すとき，読みの深さとともに模様（局面の形勢の評価）を重視することを模したプログラムである。

このように考えてくると，例え algorithm が理論的には存在する場合でも，これを用いることは実際問題として賢明でない場合があるし，更に，一般的に algorithm を求める方法は存在しないのであるから，実際の思考過程では heuristics が非常に重要な役割を果しているはずだと考えられよう。

実際，数学のように，その客観的形式においては完全に algorithmic な体系においても，数学者，あるいは数学を学ぶ者の思考過程においては，heuristics が極めて重要な役割を果している事を Polya (1953) は豊富な例をあげて論じている。普通，数学は経験科学と違って，特殊な知的体系と考えられ易い。特に，形式主義的に数学というものを捉えたとき，そのような印象をうける。しかしそれは，数学というものの客観的形式が与える印象であって，数学をつくってゆく過程は，経験科学をつくってゆく過程と本質的に共通する面を多分に持っていると言えるのではないだろうか。例えば，数学における新しい定理の発見はどのようにしてなされるであろうか。それは，しばしば，いくつかの具体例の観察に基づいて多分このような定理が成立するのではないかという予想がたてられ，その予想が公理あるいは既に証明済みの定理からの論証によって確かめられる，というような過程を経てなされるであろう。あるいは又，Polya が述べているように，数学の問題を解く際に，人々は，似たような型の問題で algorithm が知れているものを想起し，それを与えられた問題に試してみる，そしてうまくいかなかったら algorithm を少し変えてみるというようなことを屢々行なうであろう。このような思考過程は，経験科学の領域で，科学者が現象の観察や実験から法則を導いたり，あるいは，問題を解決したりする際の思考過程と頗る似ている。恐らく，両者に共通な heuristics が存在するのではないかという予想も成り立つ。このような意味で，数学ならびに諸科学において

科学者が用いて来た，あるいは現に用いている heuristics がどのようなものであるか（それは，数学並びに諸科学に共通のものもあれば，個別科学に特有なものもあるであろう）を検討することは，思考の心理学にとって甚だ興味があることと思われるのであるが，いまここでこの問題に立ち入る余裕はない。

上述のような heuristics を用いた思考の模型がいくつかつくられているが，総括的には既に Minsky (1961) による適切な批評が存在するので，ここでは，その中で最も一般的な思考模型を目指していると思われる GPS と呼ばれるプログラム（General Problem Solver. Newell, Shaw & Simon (1959), Newell & Simon (1961)）について簡単に触れ，あわせてその問題点について述べることにする。

一般に，この種の模型の検証は次のようにして行われる。模型を表わすプログラムを，特定の問題を表わすデーターと共に計算機に入れて動かし，機械の動作の系列 (trace) を記録する。一方，同じ問題を被験者に与えて解かせてみる。被験者が問題解決過程で考えることは直接には観察出来ないが，“think aloud” という教示を与えて，考えの過程を出来るだけ口に出して言わせ，それを逐一記録する (protocol)。そして，計算機の trace と被験者の protocol を比較するのである。勿論，計算機の trace は機械の言葉で書かれ，人間の protocol は普通の言葉で書かれるから，両者を比較するためには，翻訳（二つの言葉の対応づけ）が必要であるが，翻訳には多少とも恣意性が伴う。しかし，protocol に記録された言葉の細かいニュアンスは無視して，どういう手順で考えが進められたかということだけをとり出せば，これを計算機の trace と比較することが出来るであろう（更に，“think aloud” という教示によってどの程度思考過程を捉えることが出来るかという問題もある。しかし，この問題は理論的に論ずるよりもむしろ，このような実験の積み重ねにもとづいて経験的に明らかにすべきことであろう）。このようにして，いろいろな問題について trace と protocol を比較して，もし良い一致が認められれば，問題のプログラムを思考の模型と呼ぶことが出来る，というわけである。Newell, Shaw & Simon は，簡単な記号論理の問題，一種のパズル，簡単な代数や三角の問題などについて GPS を検証し，trace と protocol の間に基本的にはよい一致を見出したことを報告している。

さて，Newell, Shaw & Simon によれば，GPS のプログラムは，基本的には極めて簡単であって，それは次のようなものであると言う。任意の問題場面において，初期状態も目標状態も既に記号化されており，更に，その場面で

可能な行動はすべて，状態を表わす記号列の変形規則として与えられているということを前提とする。この前提の下で，GPS はまず，初期状態 a と目標状態 b を比較し，その間の差 d（記号配列の違い）を検出する。ついで，d を減らすような効果をもち，かつ a に適用可能な変形規則 q を与えられた規則の中から探し出す。次に，q を a に適用し，その結果得られた状態 a′ を b と比較する。このような手順を繰返すというのが GPS のプログラムである。

しかしながら，Minsky (1961) が適切に指摘しているように，上のプログラムは要するに状態間の差を減らすことによって目標に近づくということであるが，初期状態と目標状態との差を減らすということは，問題そのものなのであって，問題解決の仕方ではない。つまり，どのような差に対してどのような規則を適用するかということが指定されなければプログラムは完成しないはずである。この点について Newell, Shaw & Simon は余り明確に述べていないが，例えば，状態間のあらゆる差をあらかじめ適当に分類しておき，それぞれの差に対して関係のある変形規則を書きならべたリストをつくり，これをプログラムの中に組み込む，というようなことを行なっている模様である。

確かに，状態間のどのような差に対してどのような規則を適用するかという判断が思考の一つのかなめであることは明らかであるが，GPS のように，これを問題毎にあらかじめリストにしてプログラムに組み込むのでは，GPS の自称する generality は大分損なわれるのではないかと思われる。general な問題解決プログラムと言い得るためには，困難なことではあろうが，規則適用の heuristics（どのような差にどのような変形規則を適用したらよいかを一般的に決める方法）をつくって，これをプログラムに組み込むことが必要であろう。

この点に関して次のことが指摘される。思考というような高等な精神機能を計算機で simulate するということは，しばしば人に奇異と驚嘆の念，そして時には嫌悪の念を抱かせ易い。しかし，注意しなければならないのは，計算機が simulate すると言っても，計算機の機械装置が人間的な働きをするわけではなく，その本質はプログラムにあるということである。そして，プログラムというのは，前述のように，本質的には Turing 機械の動作表であるから，本来言葉によっても記述可能な命題を記号化したものである。そこで，計算機による simulation 模型は常に言葉による命題に引き戻してみることが出来る。ところで，言葉による一般命題は，その中に含まれる特殊命題の数を増やせば増やす程，一般的に，数多くの現象を説明することが出来るという性質をもっ

ている。勿論，説明というからには，特殊命題の単なる集まりであってはならず，多少とも一般性がなくてはならないが，特殊命題を多く含ませる程，説明力，予測力が増大するという事情は，数量的模型で函数のパラメターの数を増すと予測力が増大する事情と対応している。つまり，理論は，単に説明力，予測力の大きさによっては評価出来ないのであって，その理論のもつ一般性と対比して説明力，予測力が見られねばならないのである。このことは，言葉で記述された理論や数量的模型では全く明らかなことであるが，言葉でも数量でもない「計算機の言葉」で書かれた模型になると，計算機というものが一般の人に接近しにくいものだけに，とかく不明瞭になり勝ちである。GPS で，前述の差の種類と規則との関係を示すリストは，いわば一般命題の中の特殊命題にあたる。このようなものを任意につけ加えていくと，理論として余り意味のないものになってしまう恐れがある。計算機による simulation 模型は，数量化という制限をはずし，同時に言葉というものにつきものの恣意性を排した点で大きな可能性を開いたと言えるが，それだけに，別の面から恣意性が入ってくることがないように注意が必要であろう。

§5 今後の問題

上述のような問題はあるが，GPS はこの方面における pioneer work として高く評価されるべきであろう。しかし，これによって，思考の模型化が大きく前進したというより，むしろ可能性の開拓が漸く始まったというべきである。残された問題は甚だ多いが，紙数も尽きたので，要点だけを簡単に記す事にする。

1. Newell, Shaw & Simon は問題状況というものを記号配列およびその変形として捉えている。これは，前節までに述べたように，思考の模型化を考えるとき一つの自然な接近の仕方であると思うが，彼らが問題状況の記号化を既に与えられたものとして出発していることには問題がある。GPS は記号論理や幾何学の問題，あるいはパズルの問題など，既に記号化された問題，または容易に記号化され得る問題に適用されて一応の成功を収めているが，すべての問題が記号化されているとは限らない。いやむしろ，記号化されていない問題の方が遙かに多いであろう（問題が言葉で記述されたとき，言葉は広い意味の記号であるが，ここで言う記号とは，algorithm や heuristic の対象となり得る程明確に操作の規則が規定されるものを指す。言葉は，かかる狭い意味では未だ記号ではない）。そこで，問題状況を如何にして記号化するかという問

題が生ずるが，実は，これは思考の問題と無関係ではない。問題状況の記号化の仕方が適切であるか否かによって，問題解決の成功，不成功が大きく左右されることが多いからである。屡々，問題の適切な定式化（問題状況の記号化）に成功したら，その問題は半ば以上解けたと言える，というようなことが言われるが，これは，この間の事情をあらわしたものであろう。勿論，ある定式化が適切であるか否かは，最終的には，それによって問題がうまく解けるか否かによって判定される。そこで，実際の思考過程においては，ある記号化を採用して問題解決を試み，それでうまくいかなかったら，また別の記号化を試みるといったように，ある程度ゆきつ戻りつの試行錯誤が行われるであろう。ただし，不適切な定式化が問題場面の分析の不足に起因する場合も屡々あると思われる。何故なら，適切な記号化を行うためには，まず与えられた場面を分析して，問題解決に関連すると思われる要因を選び出さねばならないが，この分析が不十分であると，適切な記号化は期待し難いからである。ソビエトの心理学者ルビンシュテイン (1958) は，問題場面の分析が問題解決に如何に影響するかと言うことについて，興味ある実験をいくつか紹介しているが，このような実験によって明らかにされる問題場面分析の heuristics や記号化の heuristics を取り入れていくことが，今後の思考模型に課せられた課題の一つであろう。

2. このことと関連するが，問題状況の不適切な定式化は，屡々，問題場面のもつ知覚的構造にも由来する。このことは Gestalt 心理学の夙に指摘したところであるが，最近でも例えばルビンシュテインは，6本のマッチ棒で辺がマッチ棒の長さに等しい四つの三角形をつくるという課題解決の実験を分析して，この課題の困難は専ら，被験者が三角形を平面上につくろうとすることに由来することを指摘している。つまり，問題のマッチ棒が平面上に与えられているために，被験者は暗々裡に目標の三角形も平面上にあることを前提とするのである。このように，問題場面を，その知覚構造ないし常識的な概念構造が問題状況の誤認に導くように作成することによって問題を困難化するというやり方は，パズルや推理小説が屡々用いる手法である。しかし，パズルや推理小説のように意識的にゆがめてつくった問題場面に限らず，ここで立ち入る余裕はないが，自然科学の歴史においても，知覚あるいは固定化した概念が問題解決を妨げた例は甚だ多い。常識で当然とされる前提を疑ってみることが科学において常に大切であるとされる所以であろう。このような知覚や概念形成と問題解決との関連がどうなっているのかという問題はなかなか難しい問題ではあるが，やはり将来の思考模型がとり組むべきことの一つであると思われる。

3. GPS では，前述のように，状態間の差を減らすということが中心的な方法であるが，ある変形規則を適用して果して差が減少したか否かの判定は，状態間の類似性が増大したか否かによってなされる。また，ある状態に，ある変形規則を適用し得るか否かの判定も，変形規則の含む項と状態との類似を手がかりとして行われる。しかし，実際の思考過程においては，このような与えられた問題内の類似にとどまらず，問題間の類似，algorithms 間の類似，heuristics 間の類似も広く利用されるであろう。一般に，類似を手がかりにして考えを進める事を類推と呼ぶとすれば，日常的思考は勿論のこと，科学的思考においても類推は極めて重要な働きをしているのではないであろうか。屢々，逆に，類推を排することこそ科学的であると考えられ易いが，科学が類推を排するとは類推を論証と混同することを排する——つまり類推によって得た仮説は論証によって確かめられねばならない——という意味であって，科学者の思考過程に類推が入ってはならないという意味では決してないであろう。実際，数学のように厳密を生命とする知的活動においてさえ，類推がいかに重要な働きを果しているかを Polya (1953) は豊富な例をあげて説明している。しかし勿論，あらゆる類推が一様に思考の有力な手段であると言うわけではない。素朴な知覚的レベルでの類似に基づく類推からはじまって，高度の概念的レベルでの類似に基づく類推まで，種々のレベルの類推が存在するであろう。それに対応して，類推の heuristics も多様なものを考える必要があると思われる。

科学においては，屢々模型を使っての推論が行われる。模型とは何かという事を一般的に定義するのは難しいが，大雑把に言うと次のようなことであろう。ここにまだその構造がよく分からない対象あるいは現象 A があるとする。別に，その構造なり法則性なりがよく解明された対象，現象 B があって，これが A と何かある本質的な点で類似しているように思われるとする。そのとき，この類似点を手がかりにして類推を行い，B の構造，法則を模してつくった A に関する仮説，それが模型であると言えるのではないであろうか。例えば，ここに述べた思考模型は，前述のように，Turing 機械という，その構造がよく分かったものの働きと人間の思考過程との間に一種の類似を認め，それに基づいてつくられた仮説である。また，Rutherford の原子模型は，周知のように，古典力学でよく解明された太陽系の運動になぞらえてつくられた仮説であった。勿論，模型をつくる際には当然いくつかの非類似点が無視されよう——それが後になって重要である事が分かる，ということは常に起こり得る。実際，Rutherford の原子模型は後に量子力学によって訂正されねばならなかったの

である。しかしながら，この模型の果した heuristic な役割りは甚だ大きかったと言わねばならない（heuristic という言葉は，勿論，元来，“発見を助ける”という意味であるが，実際，Rutherford の模型は原子構造に関する思考を一歩進め，数多くの実験を刺激し，発見を生んだと云われる）。

このような，模型による類推という heuristics も，思考模型にとって，困難な，しかし興味ある課題であろう。

（後記）

このあと，帰納的推論について述べる積りであったが，与えられた紙数が尽きたので，別の機会に取り上げることとしたい。

引　用　文　献

Bush, R. R. & Mosteller, F. 1955 *Stochastic models for learning.* Wiley.

Feigenbaum, E. A. & Feldman, J. 1963 *Computers and thought.* McGraw-Hill.

Kleene, S. C. 1952 *Introduction to metamathematics.* NorthHolland Publishing Co.

Luce, R. D., Bush, R. R. & Galanter, E. (eds.) 1963 *Handbook of mathematical psychology.* vol. 1. Wiley.

Minsky, M. 1961 Steps toward artificial intelligence. *Proceedings of the Institute of Radio Engineers,* 49, 8-30.

Newell, A., Shaw, J. C., & Simon, H. A. 1959 Report on a general problem-solving program. *Proceedings of the international conference on information processing,* UNESCO, 256-264.

Newell, A. & Simon, H. 1961 GPS, a program that simulates human thought. *Lernende Automaten,* Munich: R. Oldenbourg KG. Also reproduced in in Feigenbaum & Feldman (1963).

Piaget, J. 1953 *Logic and psychology.* Manchester Univ. Press.

Polya, G. 1953 *Mathematics and plausible reasoning.* vol. I & II. Princeton Univ. Press. (柴垣訳. 数学における発見はいかになされるか. I, II. 丸善)

ルビンシュテイン（石田訳），1958 思考心理学—その研究方法，明治図書，1962

赤　摂 也，1963，基礎論，現代の数学，第6分冊，ダイヤモンド社

Shannon, C. E. & McCarthy, J. (eds.) 1956 *Automata studies.* Princeton Univ. Press.

Turing, A. M. 1936-7 On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society* (2), *42,* 230-265. A correction, *ibid., 43,* 544-546. Also in Goodman, R. (ed.) *Annual Review of Automatic Programming.* Pergamon Press, 1960.

Résumé

Psychology and Mathematical Models

— Particularly on Non-numerical Models of Thought —

Yoichiro Takada

This article is an attempt to analyse some basic problems which underlie mathematical models in psychology, with a special emphasis on non-numerical models (computer simulation models) of thought. First, the meaning of "mathematical" model-building was briefly discussed, and the logical structure of deductive reasoning was examined, which then led to the analysis of the relationship of "Turing" machine to the problem solving in general. These analyses point out that models of thought must somehow incorporate general *heuristics*. One such attempt is the GPS program by Newell, Shaw & Simon, which is a suggestive pioneer work in this field, yet embodies only a limited range of heuristics supposedly used in our daily, as well as scientific, reasoning. A few of these heuristics were roughly discussed, with the hope that they will be incorporated in the future programming of thought process.